APPENDICES A & B

United States Patent Application

for

METHODS AND APPARATUS FOR IDENTIFYING RELATED NODES IN A DIRECTED GRAPH HAVING NAMED ARCS

Appendix A

RDF Rules:
==========

From the following definitions:

Any nodes matching criteria are valid nodes.
All ancestors of a valid node are valid.
All decendants of a valid node are valid, unless they share a predicate with an already valid decendant.

We get the following rules:

Criteria Rules
- Match criteria to triple facts in knowledge base.
- Literals are matched on predicate and object (literal)
- Resources are matched on predicate (only if supplied in criteria) and object (resource)

Sibling Rules
- Finds other valid triple facts at the same level.
- All siblings of a valid node (triples that share the same subject) are valid, unless they have the same predicate as a criteria predicate.

Ancestor Rules
- Backward chaining up the directed graph to find triple facts from valid nodes and criteria
- All ancestors (triples whose object is equal to the subject of the matched triple) of a valid triple are valid, unless there is criteria that negates the subject resource of that triple.
- i.e. Walk up the tree.

Decendant Rules
- Forward chaining down the directed graph to find triple facts from valid nodes and criteria
- Any decendents of a valid triple or root node are valid unless they share a predicate name with a matched triple or criteria.

An example:

Given the following Original Model
==========

```
<?xml version="1.0"?>

<rdf:RDF xmlns:rdf="http://www.w3.org/1999/02/22-rdf-syntax-ns#"
      xmlns="http://metatomix.com/rules_test/1.0#">

 <rdf:Description rdf:about="company://id#3">
  <customer rdf:resource="company://id#1"/>
  <customer rdf:resource="company://id#4"/>
  <customer rdf:resource="company://id#2"/>
```

```
</rdf:Description>

<rdf:Description rdf:about="company://id#1">
  <employee>Howard</employee>
  <employee>Alan</employee>
  <cto>Colin</cto>
</rdf:Description>

<rdf:Description rdf:about="company://id#4">
</rdf:Description>

<rdf:Description rdf:about="company://id#2">
  <employee>David</employee>
  <cto>Colin</cto>
</rdf:Description>

</rdf:RDF>
```

Generates the following Knowledge Base Facts:

	Subject	Predicate	Object
1	company://id#3	http://metatomix.com/rules_test/1.0#customer	company://id#1
2	company://id#3	http://metatomix.com/rules_test/1.0#customer	company://id#4
3	company://id#3	http://metatomix.com/rules_test/1.0#customer	company://id#2
4	company://id#1	http://metatomix.com/rules_test/1.0#employee	Howard
5	company://id#1	http://metatomix.com/rules_test/1.0#employee	Alan
6	company://id#1	http://metatomix.com/rules_test/1.0#cto	Colin
7	company://id#2	http://metatomix.com/rules_test/1.0#employee	David
8	company://id#2	http://metatomix.com/rules_test/1.0#cto	Colin

Given the following criteria:

Criteria: Employee is Alan

```
<?xml version="1.0"?>
<rdf:RDF xmlns:rdf="http://www.w3.org/1999/02/22-rdf-syntax-ns#"
      xmlns="http://metatomix.com/rules_test/1.0#">
  <rdf:Description rdf:about='criteria://subject'>
    <employee>Alan</employee>
  </rdf:Description>
</rdf:RDF>
```

Rules Fired:

Rule 0 - Lit Criteria: Subject: company://id#1 Predicate: employee Object: Alan
Rule 1 - Siblings: Subject: company://id#1 Predicate: cto Object: Colin
Rule 2 - Ancestors: Subject: company://id#3 Predicate: customer Object: company://id#1

Triples generated:

	Subject	Predicate	Object
	criteria://subject	http://metatomix.com/rules_test/1.0#employee	Alan
1	company://id#3	http://metatomix.com/rules_test/1.0#customer	company://id#1
2	company://id#1	http://metatomix.com/rules_test/1.0#employee	Alan
3	company://id#1	http://metatomix.com/rules_test/1.0#cto	Colin

Output RDF:

```
<rdf:RDF
 xmlns:rdf='http://www.w3.org/1999/02/22-rdf-syntax-ns#'
 xmlns:NS0='http://metatomix.com/rules_test/1.0#'
 >
 <rdf:Description rdf:about='company://id#3'>
  <NS0:customer rdf:resource='company://id#1'/>
 </rdf:Description>
 <rdf:Description rdf:about='company://id#1'>
  <NS0:employee>Alan</NS0:employee>
  <NS0:cto>Colin</NS0:cto>
 </rdf:Description>
</rdf:RDF>
```

Criteria: CTO is Colin

```
<?xml version="1.0"?>
<rdf:RDF xmlns:rdf="http://www.w3.org/1999/02/22-rdf-syntax-ns#" xmlns="http://metatomix.com/rules_test/
1.0#">

 <rdf:Description rdf:about='criteria://subject'>
  <cto>Colin</cto>
 </rdf:Description>

</rdf:RDF>
```

Rules Fired:

```
Rule 0 - Lit criteria:  Subject: company://id#1  Predicate: cto       Object: Colin
Rule 1 - Siblings:      Subject: company://id#1  Predicate: employee  Object: Alan
Rule 1 - Siblings:      Subject: company://id#1  Predicate: employee  Object: Howard
Rule 2 - Ancestors:     Subject: company://id#3  Predicate: customer  Object: company://id#1
Rule 0 - Lit criteria:  Subject: company://id#2  Predicate: cto       Object: Colin
Rule 1 - Siblings:      Subject: company://id#2  Predicate: employee  Object: David
Rule 2 - Ancestors:     Subject: company://id#3  Predicate: customer  Object: company://id#2
```

Triples generated:

Subject	Predicate	Object
criteria://subject	http://metatomix.com/rules_test/1.0#cto	Colin

1	company://id#3	http://metatomix.com/rules_test/1.0#customer	company://id#1
2	company://id#3	http://metatomix.com/rules_test/1.0#customer	company://id#2
3	company://id#1	http://metatomix.com/rules_test/1.0#employee	Howard
4	company://id#1	http://metatomix.com/rules_test/1.0#employee	Alan
5	company://id#1	http://metatomix.com/rules_test/1.0#cto	Colin
6	company://id#2	http://metatomix.com/rules_test/1.0#employee	David
7	company://id#2	http://metatomix.com/rules_test/1.0#cto	Colin

Output RDF:

```
<rdf:RDF
 xmlns:rdf='http://www.w3.org/1999/02/22-rdf-syntax-ns#'
 xmlns:NS0='http://metatomix.com/rules_test/1.0#'
>
 <rdf:Description rdf:about='company://id#3'>
  <NS0:customer rdf:resource='company://id#1'/>
  <NS0:customer rdf:resource='company://id#2'/>
 </rdf:Description>
 <rdf:Description rdf:about='company://id#2'>
  <NS0:cto>Colin</NS0:cto>
  <NS0:employee>David</NS0:employee>
 </rdf:Description>
 <rdf:Description rdf:about='company://id#1'>
  <NS0:cto>Colin</NS0:cto>
  <NS0:employee>Alan</NS0:employee>
  <NS0:employee>Howard</NS0:employee>
 </rdf:Description>
</rdf:RDF>
```

Criteria: Company is Company1

```
<?xml version="1.0"?>
<rdf:RDF xmlns:rdf="http://www.w3.org/1999/02/22-rdf-syntax-ns#" xmlns="http://metatomix.com/rules_test/
1.0#">

 <rdf:Description rdf:about='criteria://subject'>
  <criteria://predicate rdf:resource="company://id#1"/>
 </rdf:Description>

</rdf:RDF>
```

Rules Fired:

```
Rule 0 - Res criteria:  Subject: company://id#3  Predicate: customer  Object: company://id#1
Rule 3 - decendents:    Subject: company://id#1  Predicate: cto       Object: Colin
Rule 3 - decendents:    Subject: company://id#1  Predicate: employee  Object: Alan
Rule 3 - decendents:    Subject: company://id#1  Predicate: employee  Object: Howard
```

Triples generated:

Subject	Predicate	Object

	criteria://subject	criteria://predicate	company://id#1
1	company://id#3	http://metatomix.com/rules_test/1.0#customer	company://id#1
2	company://id#1	http://metatomix.com/rules_test/1.0#employee	Howard
3	company://id#1	http://metatomix.com/rules_test/1.0#employee	Alan
4	company://id#1	http://metatomix.com/rules_test/1.0#cto	Colin

Output RDF:

```
<rdf:RDF
 xmlns:rdf='http://www.w3.org/1999/02/22-rdf-syntax-ns#'
 xmlns:NS0='http://metatomix.com/rules_test/1.0#'
>
 <rdf:Description rdf:about='company://id#3'>
  <NS0:customer rdf:resource='company://id#1'/>
 </rdf:Description>
 <rdf:Description rdf:about='company://id#1'>
  <NS0:cto>Colin</NS0:cto>
  <NS0:employee>Alan</NS0:employee>
  <NS0:employee>Howard</NS0:employee>
 </rdf:Description>
</rdf:RDF>
```

Appendix B

```
CTO is Colin

******* Criteria Model ********
<rdf:RDF
 xmlns:rdf='http://www.w3.org/1999/02/22-rdf-syntax-ns#'
 xmlns:NS0='http://m.com#'
 >
  <rdf:Description rdf:about='criteria://subject'>
   <NS0:cto>Colin</NS0:cto>
  </rdf:Description>
</rdf:RDF>
*********************************
 f-0  (MAIN::triple (obj "Alan") (objIsRes FALSE) (pred "http://m.com#employee") (subj "company://id#1") )
 f-1  (MAIN::triple (obj "company://id#2") (objIsRes TRUE) (pred "http://m.com#customer") (subj "company:
//id#3") )
 f-2  (MAIN::triple (obj "Colin") (objIsRes FALSE) (pred "http://m.com#cto") (subj "company://id#2") )
 f-3  (MAIN::triple (obj "Colin") (objIsRes FALSE) (pred "http://m.com#cto") (subj "company://id#1") )
 f-4  (MAIN::triple (obj "David") (objIsRes FALSE) (pred "http://m.com#employee") (subj "company://id#2") )
 f-5  (MAIN::triple (obj "Howard") (objIsRes FALSE) (pred "http://m.com#employee") (subj "company://
id#1") )
 f-6  (MAIN::triple (obj "company://id#4") (objIsRes TRUE) (pred "http://m.com#customer") (subj "company:
//id#3") )
 f-7  (MAIN::triple (obj "company://id#1") (objIsRes TRUE) (pred "http://m.com#customer") (subj "company:
//id#3") )
 f-8  (MAIN::criteria (obj "Colin") (objIsRes FALSE) (pred "http://m.com#cto") (subj "criteria://subject") )
For a total of 9 facts.

 FIRE 1 MAIN::lit-criteria-rule f-8, f-3
  ==> f-9 (MAIN::valid-triple (subj "company://id#1") (pred "http://m.com#cto") (obj "Colin"))
==> Activation: MAIN::siblings-rule :  f-9, f-8, f-0
==> Activation: MAIN::siblings-rule :  f-9, f-8, f-5
==> Activation: MAIN::ancestors-rule :  f-9, f-8, f-7
 <== f-3 (MAIN::triple (obj "Colin") (objIsRes FALSE) (pred "http://m.com#cto") (subj "company://id#1") )

 FIRE 2 MAIN::ancestors-rule f-9, f-8, f-7
  ==> f-10 (MAIN::valid-triple (subj "company://id#3") (pred "http://m.com#customer") (obj "company://
id#1"))
==> Activation: MAIN::decendents-rule :  f-10, f-8, f-0
==> Activation: MAIN::decendents-rule :  f-10, f-8, f-5
==> Activation: MAIN::root-node-rule :  f-10,,,
 <== f-7 (MAIN::triple (obj "company://id#1") (objIsRes TRUE) (pred "http://m.com#customer") (subj
"company://id#3") )

 FIRE 3 MAIN::root-node-rule f-10,,,
  ==> f-11 (MAIN::root-node (obj "company://id#3"))

 FIRE 4 MAIN::decendents-rule f-10, f-8, f-5
  ==> f-12 (MAIN::valid-triple (subj "company://id#1") (pred "http://m.com#employee") (obj "Howard"))
  <== f-5 (MAIN::triple (obj "Howard") (objIsRes FALSE) (pred "http://m.com#employee") (subj "company:
//id#1") )
<== Activation: MAIN::siblings-rule :  f-9, f-8, f-5

 FIRE 5 MAIN::decendents-rule f-10, f-8, f-0
  ==> f-13 (MAIN::valid-triple (subj "company://id#1") (pred "http://m.com#employee") (obj "Alan"))
```

```
 <== f-0 (MAIN::triple (obj "Alan") (objIsRes FALSE) (pred "http://m.com#employee") (subj "company:
//id#1") )
<== Activation: MAIN::siblings-rule :  f-9, f-8, f-0

FIRE 6 MAIN::lit-criteria-rule f-8, f-2
 ==> f-14 (MAIN::valid-triple (subj "company://id#2") (pred "http://m.com#cto") (obj "Colin"))
==> Activation: MAIN::siblings-rule :  f-14, f-8, f-4
==> Activation: MAIN::ancestors-rule :  f-14, f-8, f-1
 <== f-2 (MAIN::triple (obj "Colin") (objIsRes FALSE) (pred "http://m.com#cto") (subj "company://id#2") )

FIRE 7 MAIN::siblings-rule f-14, f-8, f-4
 ==> f-15 (MAIN::valid-triple (subj "company://id#2") (pred "http://m.com#employee") (obj "David"))
==> Activation: MAIN::ancestors-rule :  f-15, f-8, f-1
 <== f-4 (MAIN::triple (obj "David") (objIsRes FALSE) (pred "http://m.com#employee") (subj "company:
//id#2") )

FIRE 8 MAIN::ancestors-rule f-15, f-8, f-1
 ==> f-16 (MAIN::valid-triple (subj "company://id#3") (pred "http://m.com#customer") (obj "company://
id#2"))
 <== f-1 (MAIN::triple (obj "company://id#2") (objIsRes TRUE) (pred "http://m.com#customer") (subj
"company://id#3") )
<== Activation: MAIN::ancestors-rule :  f-14, f-8, f-1

Final Facts in Knowledge base:

f-6   (MAIN::triple (obj "company://id#4") (objIsRes TRUE) (pred "http://m.com#customer") (subj "company:
//id#3") )
f-8   (MAIN::criteria (obj "Colin") (objIsRes FALSE) (pred "http://m.com#cto") (subj "criteria://subject") )
f-9   (MAIN::valid-triple (subj "company://id#1") (pred "http://m.com#cto") (obj "Colin"))
f-10   (MAIN::valid-triple (subj "company://id#3") (pred "http://m.com#customer") (obj "company://id#1"))
f-11   (MAIN::root-node (obj "company://id#3"))
f-12   (MAIN::valid-triple (subj "company://id#1") (pred "http://m.com#employee") (obj "Howard"))
f-13   (MAIN::valid-triple (subj "company://id#1") (pred "http://m.com#employee") (obj "Alan"))
f-14   (MAIN::valid-triple (subj "company://id#2") (pred "http://m.com#cto") (obj "Colin"))
f-15   (MAIN::valid-triple (subj "company://id#2") (pred "http://m.com#employee") (obj "David"))
f-16   (MAIN::valid-triple (subj "company://id#3") (pred "http://m.com#customer") (obj "company://id#2"))
For a total of 10 facts.

******* Working Model ********
<rdf:RDF
  xmlns:rdf='http://www.w3.org/1999/02/22-rdf-syntax-ns#'
  xmlns:NS0='http://m.com#'
 >
  <rdf:Description rdf:about='company://id#3'>
    <NS0:customer rdf:resource='company://id#1'/>
    <NS0:customer rdf:resource='company://id#2'/>
  </rdf:Description>
  <rdf:Description rdf:about='company://id#2'>
    <NS0:cto>Colin</NS0:cto>
    <NS0:employee>David</NS0:employee>
  </rdf:Description>
  <rdf:Description rdf:about='company://id#1'>
    <NS0:cto>Colin</NS0:cto>
    <NS0:employee>Howard</NS0:employee>
    <NS0:employee>Alan</NS0:employee>
  </rdf:Description>
</rdf:RDF>
*******************************
```